Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Please amend claims 1, 6, 8, 9, 10, and 11 as follows:

1. (currently amended): Apparatus for providing efficient context switching between software tasks in a merged sequence processor (SP) and processor element (PE) processor environment, each software task comprising a plurality of instructions, the ~~SP-by-one-PE~~ merged SP and PE ~~array~~ processor environment configurable to be in a first array configuration or a second array configuration, the apparatus comprising:

a first set of registers stored in ~~the~~ an SP register file;

a second set of registers stored in ~~the~~ a PE register file;

an execution unit that is shared to execute SP instructions and PE instructions;

a fetch controller for fetching a plurality of instructions;

an instruction register for receiving each fetched instruction, the instruction register providing control information for the execution of a fetched instruction, the instruction register having a sequence processor/processing element (SP/PE) selection bit ~~in an~~ set by a fetched instruction whereby the SP/PE selection bit value can change with each fetched instruction; and

a processor state register having a context select bit (CSB) ~~in a processor state register~~, a specific instruction out of the plurality of instructions setting the CSB value, the CSB being independent of bits in the instruction register, the CSB value persisting between the fetched

instructions, the CSB value in conjunction with the SP/PE selection bit value selecting a context of a first software task utilizing the first array configuration or a context of a second software task utilizing a the second array configuration, the first array configuration including at least one register from the second set of registers to execute sequential instructions on the execution unit, the second array configuration including at least one register from the first set of registers to execute sequential instructions on the execution unit.

2. (canceled)

3. (original): The apparatus of claim 1 further comprising means for allowing the first set of registers to be saved and restored from memory in the background while a task is using the second set of registers in the foreground; and for allowing the second set of registers to be saved and restored from memory in the background while a task is using the first set of registers in the foreground.

4. (original): The apparatus of claim 3 wherein said means for allowing comprises a pair of background address registers to provide store and load addresses.

5. (original): The apparatus of claim 1 further comprising a plurality of execution units and a multiplexer connected to select which registers the execution units read data from and write data to, the multiplexer controlled by a logical combination of the SP/PE selection bit and the CSB.

6. (currently amended): Apparatus for providing efficient context switching between software tasks in a merged sequence processor (SP) and processor element (PE) processor environment, each software task comprising a plurality of instructions, the merged SP and PE

processor environment configurable to be in a first array configuration or a second array configuration, the apparatus comprising:

a first set of registers stored in a SP register file;

a second set of registers stored in a PE register file;

a sequence processor/processing element (SP/PE) selection bit in an instruction;

a context select bit (CSB) in a processor state register, a specific instruction out of the plurality of instructions setting the CSB, the CSB in conjunction with the SP/PE selection bit selecting a context of a first software task utilizing the first array configuration or a context of a second software task utilizing the second array configuration, the first array configuration including at least one register from the second set of registers to execute sequential instructions, the second array configuration including at least one register from the first set of registers to execute sequential instructions; ~~The apparatus of claim 1 wherein~~ and

the SP/PE selection bit is used ~~in a 1x1 array core having SP register files and PE register files~~ to determine whether ~~which register files~~ the SP's register files or the PE's registers files are to be accessed for each instruction execution when the CSB is inactive and to have both sequential and parallel instructions use the PE register files when the CSB is active.

7. (original): The apparatus of claim 1 wherein the first or second register files may comprise reconfigurable compute register files (CRF), address register files (ARF), miscellaneous register files (MRF) or a combination of CRF, ARF and MRF files.

8. (currently amended): Apparatus for providing efficient context switching between software tasks in an array of multiple processors including a sequence processor (SP) and

multiple processing elements (PE), each software task comprising a plurality of instructions, the array of multiple processors configurable to be in a first array configuration or a second array configuration, said apparatus comprising:

a first set of registers stored in a first register file for the SP;

additional sets of registers stored in a plurality of additional register files, with one of the additional sets of registers for each of the PEs;

a fetch controller for fetching a plurality of instructions;

an instruction register for receiving each fetched instruction, the instruction register providing control information for the execution of a fetched instruction, the instruction register having a sequence processor/processing element (SP/PE) selection bit ~~in an~~ set by a fetched instruction whereby the SP/PE selection bit value can change with each fetched instruction; and

a processor state register having a ~~software controllable~~ context select bit (CSB) loadable by a specific instruction, the CSB being independent of bits in the instruction register, the CSB value persists between fetched instructions, ~~in a processor state register which~~the CSB value in a logical combination with the SP/PE selection bit value reconfigures the array by selecting a first context in which the array is configured in a first configuration which provides sequential instructions to utilize one of the plurality of additional register files and parallel instructions to utilize the remaining plurality of additional register files or a second context in which the array is configured in a second configuration utilizing the first set of registers for sequential instructions and the plurality of additional register files for parallel instructions.

9. (currently amended): The apparatus of claim 8 wherein said array is a 1x2 array and said first configuration is a ~~1x2~~ 1x1 and said second configuration is a ~~1x1~~1x2.

10. (currently amended): The apparatus of claim 8 wherein said array is a 1x5 array and said first configuration is a ~~1x5~~ 2x2 and said second configuration is a ~~2x2~~1x5.

11. (currently amended): A method for providing efficient context switching in an array processor having a sequence processor (SP) and a plurality of processing elements (PEs), the sequence processor having ~~an~~ SP register file, each PE having a PE register file, the method comprising:

~~providing~~ fetching to an instruction register ~~a~~ an instruction having a sequence processor/processing element (SP/PE) selection bit value that specifies ~~in a first instruction~~ sequential processing;

setting a context select bit (CSB) in a processor state register with a specific instruction out of a plurality of instructions, the CSB being independent of bits in the instruction register, the CSB value persists between fetched instructions; ~~second instruction;~~

utilizing the SP/PE selection bit value, which can change with each fetched instruction, ~~in~~from the ~~first~~ fetched instruction in conjunction with the context select bit stored in the processor state register to determine a context for operation; and

configuring the array processing to have either a first configuration or a second configuration depending upon the context, the first configuration including ~~at least one register file of the plurality of PE register files~~ the SP register file for a sequential instruction, the second

configuration including at least one register file of the plurality of PE register files ~~the SP register file~~ for a sequential instruction.

12. (previously presented): The method of claim 11 further comprising the steps of:

identifying each PE of said array with both a virtual identifier and a physical identifier; and

identifying each PE utilizing its physical identifier in a first context and identifying each PE utilizing its virtual identifier in a second context.

13. (previously presented): The method of claim 12 wherein the first context is when the CSB bit is inactive and the second context is when the CSB bit is active.